

Simulation von Softwareprojekt-Modellen zur Ausbildung angehender Projektleiter

*Tilman Hampp, Patricia Mandl-Striegnitz
Abteilung Software Engineering, Universität Stuttgart
Breitwiesenstr. 20 - 22, D-70565 Stuttgart
(hampptn, mandlpa)@informatik.uni-stuttgart.de*

Abstract: Die Qualifikation des Projektleiters entscheidet über den Erfolg oder Misserfolg eines Projekts. Ein qualifizierter Projektleiter zeichnet sich nicht nur durch sein theoretisches Wissen aus, sondern besitzt zusätzlich umfassende praktische Erfahrung. In der universitären Ausbildung haben wir jedoch keine Möglichkeit, ausreichend viele Projekte unter möglichst realen Bedingungen durchzuführen, dass alle Studierenden praktische Erfahrungen sammeln können.

In diesem Artikel wird ein alternativer Ansatz zur Ausbildung von Projektleitern vorgestellt, der auf der Simulation von Softwareprojekten beruht. Die wesentliche Komponente dieses Ansatzes sind die zur Simulation eingesetzten Modelle von Softwareprojekten. Es werden verschiedene Modelle vorgestellt und die Erfahrungen im Einsatz dieser Modelle in der studentischen Ausbildung präsentiert.

1. Einführung

Untersuchungen zeigen die Bedeutung des Projektmanagements als Schlüsselfaktor für den Erfolg oder Misserfolg eines Projekts [ML99]. Ohne ein funktionierendes Projektmanagement ist es nicht möglich, ein Projekt selbst unter günstigen Bedingungen erfolgreich abzuschließen. Vor allem Schwierigkeiten im Verlauf des Projekts (z.B. Ausfall wichtiger Mitarbeiter) kann nur ein qualifizierter Projektleiter vorbeugen oder diese abmildern, um ein (großes) Desaster zu verhindern. Fachwissen alleine reicht nicht aus. Zusätzlich benötigt der Projektleiter praktische Erfahrung, um seine Projektmanagement-Aufgaben schnell und zuverlässig ausführen und in jeder (kritischen) Situation sinnvoll reagieren zu können.

Entsprechend hoch sind die Anforderungen an die Ausbildung im Software-Projektmanagement. Es muss sichergestellt werden, dass Projektleiter die erforderliche Qualifikation besitzen, bevor sie ihr erstes reales Projekt leiten. Diesen Anforderungen wird man in der Ausbildung nicht gerecht. Management-relevante Themen werden in vielen Informatik-Studiengängen nicht oder völlig unzureichend behandelt [Le00]. Vor allem bietet die Ausbildung an den Hochschulen meist keine Möglichkeit, praktische Projektmanagement-Erfahrung zu sammeln. Erstens werden in den Praktika die technischen Aspekte des Software Engineerings betont; die Studierenden übernehmen die Rolle des Entwicklers. Zweitens sind Umfang und Teilnehmerzahl der Projekte zu gering, als dass Projektmanagement-Aufgaben umfassend durchgeführt werden müssten und für den Projekterfolg entscheidend sind. Drittens verhindert die gegenüber der Praxis idealisierte Situation studentischer Projekte, dass die Schwierigkeiten realer Projekte in der erforderlichen Kom-

plexität auftreten. Und selbst wenn reale Bedingungen geschaffen werden, können auf Grund des entstehenden Aufwands nicht ausreichend viele Projekte durchgeführt werden, dass jeder Studierende im Verlauf seines Studiums in mindestens einem Projekt die Rolle des Projektleiters übernehmen kann.

Notwendig ist deshalb ein Ausbildungsansatz, der die Möglichkeit bietet, mit vertretbarem Aufwand wichtige Projektmanagement-Aufgaben zu trainieren und praktische Erfahrungen zu sammeln. Im SESAM-Projekt der Abteilung Software Engineering an der Universität Stuttgart wurde ein neuer Ansatz entwickelt, der auf der Simulation von Softwareprojekten beruht. Die Simulation basiert auf Modellen von Softwareprojekten. Ein Spieler in der Rolle des Projektleiters erhält die Aufgabe, das Projekt zu planen und anschließend am Simulator durchzuführen. Durch die Möglichkeit, in wenigen Stunden ein umfangreiches Projekt abzuwickeln, können in der Ausbildung mehrere verschiedene Projekte mit unterschiedlichen Anforderungen an das Projektmanagement geleitet werden. Dazu müssen verschiedene Modelle bereitgestellt werden. Je nach Vorbildung und Erfahrung der Zielgruppen werden verschiedene Lernziele festgelegt und entsprechende Modelle realisiert. In diesem Artikel werden drei Modelle vorgestellt. Für jedes Modell werden die Lernziele und die bereits erzielten Erfahrungen in der Ausbildung präsentiert.

Der Artikel ist folgendermaßen aufgebaut: In Kapitel 2 werden das SESAM-System und der gewählte Modellierungsansatz, in Kapitel 3 die SESAM-Modelle vorgestellt. Kapitel 4 fasst die Erfahrungen und Hypothesen zum Einsatz der Modelle zusammen.

2. Software Engineering Simulation durch animierte Modelle

2.1. Grundidee und Zielsetzung

Die Idee, ein Simulationssystem zur Ausbildung einzusetzen, kennen wir aus dem Einsatz von Flugsimulatoren in der Ausbildung von Piloten. Die Situation ist vergleichbar: Weder ein Pilot noch ein Projektleiter haben eine Chance, alle möglichen schwierigen Situationen real zu erproben, um zu lernen, wie man sie am Besten meistert. Beide müssen die Konsequenzen möglicher Fehler verstehen, *bevor* sie zum ersten Mal selbständig ein Flugzeug fliegen bzw. ein Softwareprojekt leiten dürfen. Der Einsatz von Simulationen bietet deshalb in beiden Fällen eine sinnvolle Alternative. Entscheidend ist, dass die Simulationen ausreichend realistisch sind, so dass die Erfahrungen mit gleichem Ergebnis in der Realität angewendet werden können.

SESAM steht für **S**oftware **E**ngineering **S**imulation durch **A**nimierte **M**odelle, d.h. der Ausbildungsansatz basiert auf der interaktiven Simulation von Softwareprojekten. Durch das System soll eine Übungsumgebung geschaffen werden, die es einem Spieler in der Rolle des Projektleiters erlaubt, ein simuliertes Projekt so zu leiten als handle es sich um ein reales Projekt. Mit einem solchen System lassen sich alle denkbaren Effekte und kritischen Situationen, die ein Projektleiter in der Praxis erleben kann, für die Spieler erfahrbar machen [Lu89]. Die Spieler haben die Chance, schwere Fehler zu machen, ohne Schaden anzurichten. Der vom Spieler gewählte Verlauf bestimmt die Ergebnisse des simulierten Projekts. Die Spieler können verschiedene Strategien ausprobieren, bis es

ihnen letztlich gelingt, ihr Projekt zum Erfolg zu führen.

Die Simulation basiert auf Modellen von Softwareprojekten. Der Modellbauer entscheidet – abhängig von den angestrebten Lernzielen – welche Objekte eines Softwareprojekts und mögliche Beziehungen zwischen diesen Objekten jeweils mit ihren relevanten Eigenschaften modelliert werden. Lediglich die Rolle des Projektleiters wird nicht abgebildet, sondern von dem auszubildenden Projektleiter übernommen. Dazu stellt das Modell Aktionen zur Verfügung, mit denen er den Verlauf des simulierten Projekts interaktiv steuern und kontrollieren kann. Welche Änderungen des Projektzustands der Spieler durch seine Aktionen erwirkt, hat der Modellbauer in der Dynamik des Modells festgelegt.

2.2. Modellierungsansatz

SESAM-Modelle bestehen aus drei Komponenten: Schemamodell, Regelmodell und Situationsmodell [Sc94]. Im Schemamodell werden die für das Modell relevanten Objekte durch Entitätstypen und die Beziehungen zwischen diesen Objekten durch Relationstypen definiert. Eigenschaften der Objekte und Beziehungen werden durch Attribute dargestellt, deren Typen ebenfalls im Schemamodell festgelegt werden. Während der Simulation ändert sich in jedem Simulationsschritt der Zustand des Projekts. Ein Zustand besteht aus konkreten Ausprägungen des Schemas, d.h. aus Entitäten und Relationen. Im Situationsmodell definiert der Modellbauer den Anfangszustand für die Simulation, die sog. Startsituation. Sie besteht aus den zu Beginn der Simulation vorhandenen Entitäten und Relationen mit ihren konkreten Attributwerten.

Die Änderungen des Projektzustands und damit die Dynamik werden durch Regeln im Regelmodell beschrieben. Eine Regel besteht aus einem Bedingungs- und einem Aktionsenteil. Der Bedingungsteil beschreibt, welche Entitäts- und Relationstypen existieren müssen und gibt zusätzliche Einschränkungen für die Attributwerte an. Erfüllt der Zustand diese Bedingungen, wird die Regel an den Zustand gebunden und die Anweisungen des Aktionsteils ausgeführt. Der Aktionsteil beschreibt die Änderungen am aktuellen Zustand. Es können Entitäten oder Relationen erzeugt oder gelöscht werden oder Attributwerte verändert werden. Zusätzlich können Nachrichten an den Spieler ausgegeben werden. Das SESAM-System unterstützt neben diesen diskreten Regeln auch Aktivitätsregeln. Durch Aktivitätsregeln lassen sich Aktivitäten in einem Softwareprojekt beschreiben, die mehrere Zeitschritte umfassen. Neben einem Aktivteil, der in jedem Zeitschritt ausgeführt wird, enthalten Aktivitätsregeln auch einen Aktivierungs- und Deaktivierungsteil, der zu Beginn bzw. am Ende der Aktivität ausgeführt wird. Im Regelmodell wird außerdem über Kommandos festgelegt, durch welche Aktionen der Spieler in das Projekt eingreifen kann.

Der Zustand wird in jedem Simulationsschritt neu berechnet. Dazu werden alle Regeln überprüft und ggfs. ausgeführt, bis keine Regel mehr gebunden werden kann. Am Ende eines Simulationsschritts wird die Modellzeit entsprechend der verbrauchten Zeit weitergeschaltet. Der Regelauswertungszyklus wird angestoßen, wenn der Spieler einen oder mehrere Zeitschritte weiterschaltet oder der Zeitverbrauch der ausgeführten Aktionen größer wird als die in einem Zeitschritt verfügbare Projektleiterzeit. Erst wenn das Pro-

jekt um die insgesamt verbrauchte Zeit weitergeschaltet wurde, ruht der Simulator und der Spieler kann erneut in das Projekt eingreifen.

3. Modelle zur Ausbildung angehender Projektleiter

Ein erstes umfassendes Simulationsmodell, das sog. Qualitätssicherungs-Modell (QS-Modell), wurde von [Dr00] realisiert. Dieses Modell diente als Grundlage für die Entwicklung des Modells für Fortgeschrittene und der feingranularen Variante. Daher sind allen drei bisher entwickelten Modellen einige Grundeigenschaften gemeinsam. Alle drei Modelle bilden kleine bis mittlere Auftragsprojekte mit einem Umfang von 200 bis 1200 Adjusted Function Points (AFP) und durchschnittlich ca. fünf Entwicklern ab. Sie unterstützen aktivitäten-orientierte Prozessmodelle.

Die Inhalte der Simulationsmodelle ergeben sich aus den Aufgaben des Software-Projektmanagements und den dabei auftretenden Problemen [Th97]. In allen SESAM-Modellen werden Aufgaben des Projektleiters zur Planung, Stellenbesetzung und Fortschrittskontrolle berücksichtigt. Die Organisation des Projekts ist vorgegeben, Aufgaben zur Personalführung müssen vom Spieler erst im Modell für Fortgeschrittene wahrgenommen werden. Die notwendige Planung muss der Spieler außerhalb des Modells vornehmen. Wie in realen Projekten erhält er dazu Vorgaben des Kunden in Bezug auf das Budget, den Fertigstellungstermin und die geforderte Funktionalität und Qualität.

Die bisher realisierten SESAM-Modelle bieten alle notwendigen Aktivitäten von der Analyse bis zur Auslieferung der Software an den Kunden an. Der Projektleiter muss entscheiden, welche Tätigkeiten wann und von welchen Mitarbeitern durchgeführt werden sollen. Durch die umfassende Modellierung des gesamten Entwicklungsprozesses lassen sich die Auswirkungen von Fehlern während der Projektdurchführung auf die Zwischen- und Endresultate verdeutlichen.

3.1. Modell für Einsteiger (QS-Modell)

Zielgruppe

Das QS-Modell wird eingesetzt, um Studierenden die Grundlagen des Projektmanagements praktisch zu vermitteln. Die Teilnehmer besitzen grundlegende theoretische Kenntnisse, aber keine oder nur wenig praktische Erfahrung.

Lernziele

Der Spieler sollen lernen, ein Projekt selbständig zu planen, gemäß dieser Planung durchzuführen und ggfs. korrigierend einzugreifen. In der Simulation erfahren sie die typischen Schwierigkeiten bei der Planung (z.B. mangelnde Daten vergangener Projekte) und Fortschrittskontrolle (z.B. ungenaue Statusinformationen). Indem sie verschiedene Strategien ausprobieren, lernen sie, wie man diese Probleme lösen oder zumindest abmildern kann. Beispielsweise ist eine wirksame Fortschrittskontrolle nur möglich, wenn Zwischenergebnisse regelmäßig durch Qualitätssicherungsmaßnahmen geprüft werden. Die Spieler sollen Kosten und Nutzen verschiedener Maßnahmen erfahren und lernen, diese sinnvoll einzusetzen. Der Spieler soll außerdem lernen, die Mitarbeiter entsprechend ihrer Qualifi-

kation auszuwählen und einzusetzen.

Inhalte und Modellierung

Um diese Ziele erreichen zu können, werden im QS-Modell Aktionen zum Einstellen neuer Mitarbeiter, zur Vergabe von Tätigkeiten an Mitarbeiter, zur Durchführung von Qualitätssicherungsmaßnahmen und zur Kontrolle des Projektverlaufs angeboten. Die Aktionen lösen bestimmte Effekte aus, die der Spieler in der Simulation erfahren kann. So erfährt der Spieler beispielsweise, dass ein Entwurfsspezialist gegenüber einem Berufsanfänger deutlich weniger Aufwand investieren muss, um den Entwurf zu erstellen und zudem einen qualitativ deutlich besseren Entwurf abliefern.

Der Inhalt der modellierten Dokumente besteht aus Anforderungen, die durch die Aktivitäten der Mitarbeiter im Idealfall entlang des Wasserfallmodells aus einer Vorgabe in ein Folgedokument übertragen werden. Es ist möglich, einzelne Phasen zu überspringen, so dass auch ein Code-and-Fix-Ansatz verfolgt werden kann. Die einzelnen Aktivitäten laufen ohne weitere Eingriffe des Spielers ab. Werden mehrere Mitarbeiter für die gleiche Aktivität eingesetzt, teilen sie sich die Arbeit. Durch den modellierten Effekt lernt der Spieler, dass das Ergebnis der Aktivität in diesem Fall früher vorliegt, durch den dabei entstehenden Kommunikationsaufwand jedoch der Aufwand insgesamt steigt.

Das Modell konzentriert sich im Wesentlichen auf die Effekte von Qualitätssicherungsmaßnahmen. Durch Reviews können frühzeitig Fehler entdeckt werden. Dabei wird das zu prüfende Dokument in einzelnen Sitzungen gegen das Vorgabedokument geprüft. Je erfahrener die Gutachter sind, desto mehr Fehler werden gefunden. Für die Spezifikation und das Handbuch kann der Kunde eingesetzt werden, er entdeckt zusätzliche Fehler. Versäumt es der Spieler, Reviews einzusetzen, werden (frühe) Fehler in alle nachfolgenden Dokumente übertragen und können erst im Test gefunden werden. Die Korrektur dieser Fehler ist entsprechend aufwändiger.

Das QS-Modell umfasst insgesamt mehr als 400 Regeln. Die Quantifizierung erfolgte nach [Jo96]. Wo quantitative Daten fehlen, wurden Parameter durch Hypothesen gestützt.

3.2. Modell für Fortgeschrittene (QSVA-Modell)

Zielgruppe

Angehende Projektleiter, die mit diesem Modell spielen, beherrschen die grundlegenden Aufgaben eines Projektleiters und können unter Idealbedingungen ein Softwareprojekt erfolgreich durchführen. Sie sind jedoch nicht darauf vorbereitet, mit unerwarteten Schwierigkeiten beispielsweise durch den Ausfall von Mitarbeitern umzugehen. Auch haben sie noch keine praktische Erfahrung in der Personalführung.

Lernziele

Im QS-Modell zeigen die Mitarbeiter abhängig von der Erfahrung eine gleichbleibende Leistung, die nicht durch weitere Faktoren wie beispielsweise ihre Motivation beeinflusst werden kann. In einem realen Projekt müsste ein Projektleiter aber durch die Personalführung sicherstellen, dass die Mitarbeiter motiviert sind, damit ihre Leistung nicht abfällt oder sie das Projekt aus Frustration verlassen.

Die Lernziele im Zusammenhang mit dem QS-Modell werden deshalb um Aspekte der Personalführung erweitert. Die Studierenden sollen lernen, wodurch die Motivation von Mitarbeitern beeinflusst wird, wie sich die Motivation der Mitarbeiter auf deren Leistung und das Projekt auswirken und wie sie als Projektleiter die Motivation der Mitarbeiter fördern und erhalten können. Dazu gehört auch, zu lernen, Mitarbeiter weder zu unter- noch zu überfordern, indem sie beispielsweise untätig herumsitzen bzw. über einen längeren Zeitraum Überstunden angeordnet werden. Durch das Modell werden die Folgen unvorhersehbarer Einflüsse wie beispielsweise die Krankheit oder Kündigung von Mitarbeitern gezeigt. Die Spieler sollen lernen, das Beste aus dieser Situation zu machen.

Modellerweiterungen

Um diese Lernziele zu erreichen, wurde das QS-Modell erweitert. Zentraler Aspekt des sog. QSVA-Modells (um Verhaltensaspekte erweitertes QS-Modell) ist dabei die Motivation der Mitarbeiter, die neben der Erfahrung die Leistung des Mitarbeiters beeinflusst. Ein hohes Interesse an einer Aktivität fördert die Motivation, während eine hohe Anzahl Überstunden die Motivation senkt. Der Projektleiter kann die Motivation der Mitarbeiter gezielt durch Mitarbeitergespräche oder verschiedene Maßnahmen, etwa Seminare, steigern. Er kann Mitarbeiter loben und in Statusmeetings über den Projektstand informieren.

Kündigungen durch einen Mitarbeiter treten entweder zufällig auf oder sind die Folge extrem schlechter Motivation. Der Verlust eines erfahrenen und motivierten Mitarbeiters senkt die Motivation der verbleibenden Teammitglieder. Hat der Mitarbeiter durch sein Verhalten das Projekt gebremst, wird sein Weggang positiv gesehen.

Auch die Reaktionen des Kunden beeinflussen die Motivation. Durch die Teilnahme an Prüfmaßnahmen erhält der Kunde einen Eindruck von der Qualität der Ergebnisse. Abhängig von seiner Zufriedenheit, die auch durch die Kommunikation mit beteiligten Mitarbeitern beeinflusst wird, lobt er die Projektmitarbeiter oder wird sich beschweren.

Zur Modellierung der zusätzlichen Aspekte wurde das QS-Modell um mehr als 100 Regeln erweitert. Die Quantifizierung musste im Wesentlichen auf Hypothesen gestützt werden. In der Literatur finden sich maximal qualitative Aussagen zu den modellierten Zusammenhängen, jedoch keine oder nur wenig brauchbare quantitative Daten.

3.3. Eine feingranulare Variante

Zielgruppe

Auch die feingranulare Variante richtet sich an Studierende, die bereits die grundlegenden Aufgaben des Projektmanagements beherrschen und Erfahrungen mit dem QS-Modell gesammelt haben. Sie haben bislang jedoch nicht gelernt, verschiedene Tätigkeiten im Detail zu planen. Ihnen fehlt außerdem die Erfahrung, welche Ereignisse den Tagesablauf des Projektleiters prägen.

Lernziele

Mit der feingranularen Variante werden zusätzlich zu den Lernzielen des QS-Modells folgende Ziele verfolgt: Die Studierenden sollen lernen, welche Aufgaben für den Projektleiter im Zusammenhang mit der Organisation, Durchführung und Nachbereitung techni-

scher Reviews [FLS00] anfallen. Sie sollen z.B. einzelne Sitzungen planen, die passenden Teilnehmer (Moderator, Gutachter etc.) auswählen und kontrollieren. Sie sollen erfahren, wie sich ihre Entscheidungen auf die Qualität des Reviews auswirken. Die Spieler sollen außerdem erkennen, wodurch der Tagesablauf einen Projektleiters bestimmt wird und vor allem, wie viel Zeit durch Störungen und Meetings etc. verbraucht wird.

Inhalte und Modellierung

Um diese Lernziele erreichen zu können, wurde die Simulationsschrittweite von bisher einem Tag auf zwei Stunden umgestellt, so dass der Spieler alle zwei Stunden in das Projekt eingreifen und die Effekte beobachten kann. Die Reviews aus dem QS-Modell wurden ersetzt, um die individuelle Vorbereitung der Gutachter getrennt von der Sitzung zu modellieren. Der Projektleiter bestimmt, welcher Teil eines Dokuments geprüft werden soll, wann die Sitzung stattfindet und wer am Review teilnimmt. In der Vorbereitung für ein Review entdecken die Gutachter Befunde, in dem sie den Prüfling, das Vorgabedokument und Fragenkataloge betrachten. Aspekte gliedern die Fragenkataloge und werden den Gutachtern vom Projektleiter individuell zugeteilt. Neben der Erfahrung der Gutachter bestimmt der vom Projektleiter festgelegte Umfang des Prüflings die Dauer und das Ergebnis der Vorbereitung. Die Dauer der Sitzung hängt von der Anzahl Befunde und der Erfahrung des Moderators als Leiter der Sitzung ab. Ein erfahrener Moderator verhindert, dass die Gutachter den Autor des Prüflings persönlich angreifen. Nach der Sitzung wird eine Empfehlung über die Annahme des Prüflings abgegeben.

Der Tagesablauf des Projektleiters wird durch Störungen geprägt, beispielsweise berichtet der Moderator von den Ergebnissen des Reviews. Nimmt der Kunde teil, bewertet er die Qualität des Prüflings und beschwert sich ggfs. beim Projektleiter. Beschwerden, Lob und Störungen durch unerfahrene Mitarbeiter oder Treffen mit den Mitarbeitern kosten den Projektleiter wertvolle Zeit, die nicht zur Projektführung zur Verfügung steht.

Anders als im QS-Modell kann ein Mitarbeiter jetzt mehrere Aktivitäten gleichzeitig wahrnehmen. Er wählt zufällig alle zwei Stunden eine andere Aufgabe, so dass eine stärkere Führung und Kontrolle durch den Projektleiter notwendig wird.

Zur Entwicklung der feingranularen Variante auf Basis des QS-Modells mussten mehr als 100 weitere Regeln implementiert werden. Die Quantifizierung feingranularer Effekte erwies sich als äußerst schwierig, weil Daten in der Literatur auf einer gröberen Granularitätsstufe angegeben werden. Lediglich Reviews wurden bereits ausreichend feingranular untersucht, so dass sich die Modellierung zunächst auf diese Aktivität konzentriert hat.

4. Einsatz der Modelle in der Ausbildung

Die Untersuchungen von [No99] bestätigen, dass die Simulation von Softwareprojekten nicht unabhängig von weiteren Schulungskomponenten in der Projektmanagement-Ausbildung eingesetzt werden kann. Um einen Lernerfolg zu erzielen, wurde deshalb von [Ma01] ein Schulungskonzept entwickelt. Im Rahmen einer Einführungsveranstaltung werden zunächst die Bedienung des SESAM-Systems und das verwendete Simulationsmodell erläutert. Anschließend erhalten die Studierenden Informationen, die sie zur Planung ihres Projekts benötigen (beispielsweise zum geschätzten Projektumfang, zum vor-

gegebenen Fertigstellungstermin und Budget und zu den Qualitätsanforderungen). Nach der Planung sollen sie ihr Projekt am Simulator durchführen. Dabei sind sie auf sich allein gestellt, denn nur durch eigenverantwortliches Handeln können sie die erzielten Ergebnisse als Folge ihrer Entscheidungen begreifen und die notwendigen Erfahrungen sammeln. Nach Abschluss der Simulation werden die Projektverläufe und die erzielten Resultate von einem Tutor ausgewertet, um den Studierenden Rückmeldung zu ihren Stärken und Schwächen bei der Projektdurchführung geben zu können. Die Analyse der Projektverläufe zusammen mit einem Tutor ist entscheidend, um die angestrebten Lernziele zu erreichen. Nur so können die Studierenden die komplexen Zusammenhänge in einem Softwareprojekt erkennen und Lehren aus ihren Fehlern ziehen. In einer wiederholten Simulation des durchgeführten Projekts erhalten sie anschließend die Möglichkeit, das Gelernte umzusetzen. Sie können ein anderes Vorgehen wählen und in Problemsituationen alternative Lösungsstrategien ausprobieren. Bessere Projektergebnisse im zweiten Projektverlauf verdeutlichen den Studierenden den erzielten Lernerfolg.

4.1. Erfahrungen mit dem QS-Modell

Um zu untersuchen, ob sich die mit dem QS-Modell angestrebten Lernziele erreichen lassen, wurden zwei Experimente mit insgesamt 16 Teilnehmern durchgeführt. Außerdem wurde im Rahmen einer Lehrveranstaltung im 3. Semester des Studiengangs Softwaretechnik an der Universität Stuttgart eine SESAM-Schulung durchgeführt. An dieser Schulung nahmen insgesamt 39 Studierende teil. Sowohl in den Experimenten als auch in der Schulung wurde das beschriebene Schulungskonzept angewendet.

In allen Fällen sollte das QS-Modell in einer Ausprägung mit einem Umfang von 200 AFP durchgeführt werden. Dafür standen 450.000 DM und neun Monate Gesamtlaufzeit zur Verfügung. Code und Handbuch sollten je mindestens 95% der Anforderungen enthalten. Der Kunde forderte, dass im Code maximal zwölf Fehler/KLOC und im Handbuch maximal 0,5 Fehler/Seite enthalten sein dürfen.

Zur Bewertung des Lernerfolgs wurden unter anderem der Projektverlauf und die dadurch erzielten Ergebnisse im ersten und zweiten Spiel miteinander verglichen. Eine ausführliche Beschreibung der Auswertungskriterien findet sich in [Ma00].

Sowohl in den Experimenten als auch im Rahmen der Lehrveranstaltung ließ sich die Mehrzahl der aufgestellten Lernziele erreichen. In der Wiederholung wurden wesentlich bessere Projektergebnisse erzielt als im ersten Spiel. In der Schulung verbesserten beispielsweise 35 der 39 Teilnehmer ihr Gesamtergebnis, immerhin 17 Studierende verbesserten sich in Bezug auf *alle* Zielvorgaben. Vor allem die Qualität der erstellten Dokumente konnte deutlich gesteigert werden. Während der Code im ersten Spiel durchschnittlich 18,4 Fehler/KLOC enthielt, wurden im Wiederholungsspiel maximal 10,7 Fehler/KLOC ausgeliefert. Während im ersten Versuch nur zwei der 39 Schulungsteilnehmer alle Zielvorgaben (Termin, Budget und Qualitätsvorgaben) erfüllen konnten, gelang dies im zweiten Versuch immerhin 17 Spielern.

Die besseren Resultate sind wesentlich darauf zurückzuführen, dass die Studierenden die Bedeutung der Qualitätssicherungsmaßnahmen erkannt und im gesamten Projektverlauf

konsequenter auf die Qualität der Dokumente geachtet haben. Im ersten Projektverlauf wurden frühe Reviews oft vernachlässigt oder viel zu spät durchgeführt, wenn bereits Folgedokumente erstellt und somit viele Fehler übertragen worden waren. Erst am Ende des Projekts wurde dann durch aufwändige Tests versucht zu retten, was zu retten ist. Außerdem haben sie gelernt, die Organisation und Durchführung der Maßnahmen deutlich zu verbessern (beispielsweise durch die Wahl erfahrenerer Gutachter oder die Beteiligung des Kunden). Im zweiten Spiel haben die Projektleiter verstärkt darauf geachtet, die Mitarbeiter entsprechend ihrer Qualifikation auszuwählen und einzusetzen (gerade dieser Punkt bereitete ihnen allerdings auch im zweiten Versuch noch immer Schwierigkeiten). Probleme hatten die Spieler nach wie vor mit der Aufgabe, den Überblick über ihr Projekt zu behalten und den Prozess zu beherrschen. Bei der Fortschrittskontrolle zeigten sich im zweiten Spiel die größten Defizite. Informationen aus dem Projekt wurden zu selten und unregelmäßig erhoben und kaum genutzt, um Fehler oder Abweichungen von der Planung zu erkennen und steuernd einzugreifen. Obwohl viele Teilnehmer in einer Befragung zum Projektmanagement bereits vor dem ersten Spiel hervorhoben, wie wichtig es ist, den Projektfortschritt regelmäßig zu kontrollieren und den Überblick über das Projekt zu behalten, gelang es kaum jemandem, dieses theoretische Wissen im ersten Projekt umzusetzen. Die Spieler haben das Problem erkannt und in einer nach dem ersten Spiel durchgeführten Befragung nochmals verstärkt darauf hingewiesen. Jedoch gelang auch im zweiten Projekt nur wenigen eine Verbesserung. An diesem Beispiel zeigt sich zweierlei: Erstens reicht theoretisches Wissen alleine nicht aus, um in der Praxis erfolgreich zu sein. Zweitens sind mehrere Versuche notwendig, um zu lernen, wie das Wissen praktisch umgesetzt werden kann und um die notwendige Erfahrung zu gewinnen.

Eine Befragung aller Teilnehmer nach ihrer subjektiven Bewertung des Lernerfolgs bestätigt die Messergebnisse. Alle Experimenteilnehmer und mehr als 90% der Schülerteilnehmer bewerten den Lernerfolg als positiv. Die meisten konnten sich mit der Rolle des Projektleiters identifizieren und gaben an, Spaß bei der Durchführung des Projekts gehabt zu haben. Sie sehen in der Simulation eine sinnvolle Ergänzung zur konventionellen Ausbildung, um die Praxis erproben und Gelerntes anwenden zu können.

4.2. Hypothesen zum Einsatz des QSVA-Modells

Das QSVA-Modell wurde bislang nicht in der Lehre eingesetzt. Es liegen jedoch Erfahrungen aus der Validierung des Modells gegenüber dem QS-Modell vor. Die dabei erzielten Ergebnisse sind insgesamt plausibel. Außerdem bestätigen die Ergebnisse eines Studenten, den wir gebeten hatten, mit dem Modell zu experimentieren, unsere Vermutungen: Obwohl der Student bereits Erfahrungen mit dem QS-Modell sammeln konnte und dabei durchaus gute Ergebnisse erzielt hat, scheiterte er im QSVA-Modell an den zusätzlichen Anforderungen. Er hatte große Schwierigkeiten, die Mitarbeiter so einzusetzen und zu führen, dass sie motiviert waren.

Wir gehen davon aus, dass es erforderlich ist, im ersten Spiel mit dem QSVA-Modell zunächst nicht alle realisierten Effekte einzusetzen, sondern den Spieler in mehreren Spielverläufen nach und nach an die Komplexität der Personalführung heranzuführen. Ansonsten ist der Spieler nicht nur völlig überfordert, sondern es wird auch schwierig

sein, ihm die Komplexität in der Analyse zu vermitteln. Die verschiedenen Effekte können durch Parameter in der Startsituation aktiviert oder deaktiviert werden.

4.3. Hypothesen zum Einsatz der feingranularen Variante

Auch die feingranulare Variante wurde bislang noch nicht in der Lehre eingesetzt. Es liegen aber ebenfalls Erfahrungen aus dem Test des Modells und der Validierung des Modells gegenüber dem QS-Modell vor.

Um den Nutzen, aber auch die Probleme beim Einsatz der Variante in der Ausbildung zu untersuchen, wurden Hypothesen aufgestellt und anhand von Testfällen untersucht. Zum Nutzen zählen die folgenden Hypothesen: *Der Spieler kann die Entwickler gezielter steuern und beobachten.* Einzelne Aktivitäten der Entwickler wie die Vorbereitung auf eine Reviewsitzung oder die daraufhin notwendige Korrektur dauern (auch im QS-Modell) nur wenige Stunden. Erst durch die feingranulare Modellierung kann der Spieler diese Tätigkeiten jedoch alle zwei Stunden kontrollieren und an einem Tag verschiedene Tätigkeiten anweisen. *Der Spieler kann erfahren, wodurch der Tagesablauf eines Projektleiters bestimmt wird.* Da der Spieler mehrmals am Tag eingreifen kann (und muss) und auch entsprechend häufig Nachrichten aus dem Projekt erhält, wird der Zeitverbrauch durch eigene Aktionen, aber auch durch Störungen deutlicher.

Neben diesem Nutzen werden aber auch Probleme erwartet. *Der Spieler braucht länger, um ein simuliertes Projekt durchzuführen.* Es ist zu vermuten, dass bei einem Zwei-Stunden-Takt mehrere Tage zur Simulation eines Projekts aufgewandt werden müssen (statt weniger Stunden, die zur interaktiven Simulation des QS-Modells benötigt werden). *Der Auswertungsaufwand für den Tutor wird größer.* Erstens hat der Spieler einen größeren Handlungsspielraum und kann dadurch mehr Fehler machen. Diese Fehler müssen in der Analyse eines Spiels erkannt und in der Diskussion mit dem Spieler erklärt werden. Zweitens sind die komplexen Zusammenhänge des QS-Modells in der feingranularen Variante erweitert worden und noch schwerer zu durchschauen. Der Spieler wird deshalb um so mehr auf die Unterstützung des Tutors angewiesen sein.

5. Offene Fragen

Die bisher erzielten Ergebnisse beim Einsatz des SESAM-Simulationssystems in der Ausbildung bestätigen, dass sich damit die gewünschten Lernerfolge erzielen lassen. Experimente mit dem QSVA-Modell müssen zeigen, ob sich auch mit einem komplexeren Modell die Lernziele erreichen lassen. Sind Spieler in der Lage, die in diesem Modell abgebildete Komplexität zu beherrschen und ist es auch dem Tutor möglich, die Stärken und Schwächen bei der Projektdurchführung zu analysieren? Da auch in diesem Modell noch einige Effekte realer Softwareprojekte fehlen, schließt sich daran die Frage an, ob es überhaupt möglich ist, in einer Simulation die Komplexität eines Softwareprojekts vollständig abzubilden und zu unterrichten. Dabei muss auch untersucht werden, welche Komplexität für Fortgeschrittene sinnvoll zu wählen ist, um die Spieler weder zu unterfordern noch einen Lernerfolg durch zu große Komplexität zu verhindern.

Im Zusammenhang mit der feingranularen Variante gilt es zu untersuchen, bis zu welcher Granularität sich ein vollständiges Projekt sinnvoll in einer Schulung durchspielen lässt. Außerdem ist es fraglich, ob alle Prozessaktivitäten ausreichend feingranular modelliert werden können, weil anders als bei technischen Reviews entsprechende Informationen in der Literatur fehlen. Wir vermuten, dass sich eine feingranulare Variante sinnvoll nur zur Schulung einzelner Projektphasen einsetzen lassen wird.

Eine andere wichtige und bislang nicht beantwortete Frage ist, ob die mit SESAM geschulten Projektleiter tatsächlich erfolgreicher sind, wenn sie ihr erstes reales Projekt leiten. Können sie die Erfahrungen aus der Simulation auf die Realität übertragen? Die Untersuchung dieser Frage würde mehrere Jahre umfassen. In dieser Zeit würden so viele zusätzliche Effekte auf die Versuchspersonen einwirken (Weiterbildungsmaßnahmen, Erfahrungen als Entwickler in realen Projekten etc.), dass eindeutige Untersuchungsergebnisse nicht zu erwarten sind. Da unsere bisherigen Erfahrungen aber zeigen, dass es durch SESAM gelingt, den Studierenden die Aufgaben und Probleme des Projektmanagements bewusst zu machen und Verbesserungen im Vorgehen zu erzielen, sind wir zuversichtlich, dass diese Erfahrungen auch auf reale Projekte übertragen werden.

Literaturverzeichnis

- [Dr00] Drappa, A.: Quantitative Modellierung von Softwareprojekten. Shaker, Aachen, 2000.
- [FLS00] Frühauf, K.; Ludewig, J.; Sandmayr, H.: Software-Projektmanagement und -Qualitätssicherung. 3. Auflage, vdf, Zürich, 2000.
- [Jo96] Jones, C.: Applied Software Measurement - Assuring Productivity and Quality. 2. Auflage, McGraw-Hill, New York, 1996.
- [Le00] Lethbridge, T.: Priorities for the Education and Training of Software Engineers. Journal of Systems and Software. Nr. 1, Band 53, 2000; S. 53-71.
- [Lu89] Ludewig, J.: Modelle der Software-Entwicklung – Abbilder oder Vorbilder? Software-technik-Trends, 9 (3), 1989; S.1-12.
- [Ma00] Mandl-Striegnitz, P.: Untersuchung eines neuen Ansatzes zur Projektmanagement-Ausbildung. In (Dumke, R.; Lehner, F. Hrsg.): Software-Metriken – Entwicklungen, Werkzeuge und Anwendungsverfahren, Deutscher Universitätsverlag, Wiesbaden, 2000; S. 111-134.
- [Ma01] Mandl-Striegnitz, P.: Qualifizierte Software-Projektmanager durch simulationsbasierte Ausbildung. In (Lichter, H.; Glinz, M. Hrsg.): Software Engineering im Unterricht der Hochschulen/SEUH - Zürich 2001. dpunkt.verlag, Heidelberg, 2001; S. 23-37.
- [ML99] Mandl-Striegnitz, P.; Lichter, H.: Defizite im Software-Projektmanagement – Erfahrungen aus einer industriellen Studie. Informatik/Informatique - Zeitschrift der schweizerischen Informatikorganisationen, 5 (5), 1999; S. 4-9.
- [No99] Notter, A.: Eine Untersuchung zur Wirksamkeit der Projektmanagement-Ausbildung am Simulator. Diplomarbeit Nr. 1724, Institut für Informatik, Universität Stuttgart, 1999.
- [Sc94] Schneider, K.: Ausführbare Modelle der Software-Entwicklung. vdf, Zürich, 1994.
- [Th97] Thayer, R.: Software Engineering Project Management. In (Thayer, R. Hrsg.): Software Engineering Project Management. 2. Auflage, IEEE Computer Society Press, Los Alamitos, CA, 1997; S. 72-104.